Sizzle Selectors

Sizzle is a JavaScript selector library that offers powerful ways to select elements. You can select based off text contained (or not contained) within elements, the existence of child elements inside a parent, or if those elements do not exist.

You can use attribute selectors and pseudo-class selectors in both stylesheets and the Element Selector field. You can only use Sizzle selectors in the Element Selector field.

> Sizzle selectors can only be used in the Element Selector field and are not supported in CSS.

For more information about Sizzle, including the full list of selectors supported, refer to the jQuery Sizzle wiki.

:has

The `:has()` selector matches when a parent element contains at least one element that matches the specified selector. For example `div.parent:has(div.child)` selects `div.parent` only if a `div.child` element exists anywhere among its descendants, not just as a direct child.

:contains

The `:contains()` selector matches when a parent element has text which matches the string. For example `div.parent:contains(string)` selects `div.parent` only if it has text directly in it or anywhere among its descendants that matches the string. The string is case-sensitive and needs to only match part of the text content.

A `div:contains(order)` selector selects `divs` containing the text string `order`. It also matches `divs` that contain the word `border` because the string still matches part of the word.

:not

The `:not()` selector matches all elements that do not match the given selector.

You may specify that the matching selector is not a specific type. For example, `.box:not(div, p)` selects all elements with `box` class except `div` and `p` elements. You can also format the selector with spaces rather than comma-separating:

```
.box:not(div p)
```

You may specify other attributes, IDs, or classes (such as `div:not(.box)` that selects `divs` except ones with `box` class).

:not in CSS

You can use the `:not()` selector in CSS as well. In the example below, all elements with the `box` class get a red background *unless* they are `<p>` tags.

```
.box:not(p) {
  background: red;
}
```

:eq

The `:eq(n)` selector selects an element with an index number equal to `n`. Unlike pseudo-class selectors, the positional index for Sizzle begins at zero, so `.main > div:eq(2)` matches the third first-level `div` within `.main` rather than the second.

:nth

Similar to `eq`, the `:nth(n)` selector matches an element with a number *n*.

:odd and :even

Because the positional index begins at zero, `:odd` and `:even` work counterintuitively.

`li:odd`

This matches the second, fourth, sixth (and so on) list items.

`li:even`

This matches the 1st, 3rd, 5th (and so on) list items.

:gt

This selects all elements *greater than* the specified number. Remember, the positional index is 0.

`.box:gt(1)`

This would select all `.box` after the 2nd instance.

:lt

This selects all elements *less than* the specified number. Remember, the positional index is 0.

`.box:lt(4)`

This selects the first 4 instances of `.box`. Remember, 0-3 are the first 4 instances.

:first

The `:first` pseudo-class is equivalent to `:eq(0)`. You could also write it as `:lt(1)`.

The notable difference between `:first` and `:first-child` is that this *only* matches a single element. `:first-child` can match more than one element, including one for each parent.

`div p:first`

In this example, the selector matches *only* the first `p` element within a `div`.

:last

The `:last` selects a single element by filtering *the current sizzle or jQuery collection* and matching the last element within it.

> ℹ The current collection can change based on interactive elements within a page.

`div p:last`

In this example, the selector matches *only* the last `p` element within a `div` based on the current Sizzle or jQuery collection.

> ℹ When you use selectors in a global CSS stylesheet, the styling applies to all elements that they match. When used in Monetate actions in the Element Selector field, they only match the first instance unless you enable **Select Multiple**.

For a complete list of CSS selectors, refer to W3 Schools. For an example of various CSS selectors, refer to the Selector Tester application on W3 Schools.

> ℹ Some CSS selectors are not supported by earlier versions of Internet Explorer. For a more in-depth list, please refer to the compatibility chart available on Quirksmode.